Algorithmic Social Sciences Research Unit

ASSRU

Department of Economics
University of Trento
Via Inama 5
381 22 Trento Italy

Discussion Paper Series

12 – 11

THE PHILLIPS MACHINE, THE ANALOGUE COMPUTING TRADITION IN ECONOMICS AND COMPUTABILITY[♣]

K. Vela Velupillai

March 2011

♣ In recent years, troubled by the claims of many in macroeconomic theory on the importance, indeed the necessity, of numerical algorithms to solve analytically intractable problems, I have felt the need to return to the noble tradition of analogue computing. There is no better way to do this than through a thorough understanding of the meaning and aims with which Phillips constructed his hydraulic analogue computing machine. Conversations with my friend and colleague Stefano Zambelli and our graduate students, Kao Selda and V. Ragupathy have, as always, been edifying. Discussions with Brian Hayes, Allan McRobie and Michael Kuczynski were also most useful. Alas, I must bear full responsibility for all the remaining infelicities.

Abstract

In this paper I try to argue for the desirability of analog computation in economics from a variety of perspectives, using the example of the Phillips Machine. Ultimately, a case is made for the underpinning of both analog and digital computing theory in constructive mathematics. Some conceptual confusion in the meaning of analog computing and its non-reliance on the theory of numerical analysis is also discussed. Digital computing has its mathematical foundations in (classical) recursion theory and constructive mathematics. The implicit, working, assumption of those who practice the noble art of analog computing may well be that the mathematical foundations of their subject is as sound as the foundations of the real analysis. That, in turn, implies a reliance on the soundness of set theory plus the axiom of choice. This is, surely, seriously disturbing from a computation point of view. Therefore, in this paper, I seek to locate a foundation for analog computing in exhibiting some tentative dualities with results that are analogous to those that are standard in computability theory. The main question, from the point of view of economics, is whether the Phillips Machine, as an analog computer, has universal computing properties. The conjectured answer is in the negative.

JEL Codes: C02, C63, E27, E37,

Keywords: Phillips Machine, Analogue Computation, Digital Computation, Computability, General Purpose Analogue Computer

1 Introductory Notes

> "More specifically, do computer trajectories 'correspond' to actual trajectories of the system under study? The answer is sometimes no. In other words, there is no guarantee that there exists a true trajectory that stays near a given computer-generated numerical trajectory. ....
>
> Therefore, the use of an ODE solver on a finite-precision computer to approximate a trajectory of a .... dynamical system leads to a fundamental paradox. .... Under what conditions will the computed trajectory be close to a true trajectory of the model?"
>
> [35], p.961.

There are two caveats, from an analog computing point of view, that should be remembered: one, the 'computer trajectories', referred to above, are those generated by a *digital* computer; secondly, the 'fundamental paradox' should not be relevant for an *analog* computer (subject, of course, to *machine precision*, which is different from *finite-precision* in a digital computer).

But there could be other paradoxes between the mathematical theory of an analog computing machine and its theoretical limits and its actual trajectories. The aim of this paper is to pave an introductory path towards a discussion of this possible paradox, in the context of the actual functioning philosophy and epistemology of a Phillips Machine[1].

There is a clear acknowledgement – albeit somewhat belated – to the long and rich tradition of considering computability and complexity of models defined over $\Re$, in a variety of recursive and computable analytic frameworks, by Smale, a leading advocate of side-stepping the Turing model of computation for computation over the reals, in [38], p.61:

> "Indeed, some work has *now been done* to adapt the Turing machine framework to deal with real numbers........ Thus, the foundations are probably being laid for a theory of computation over the real numbers."

This acknowledgement comes in 1991 after almost six decades of work by recursive and computable analysts to *adapt* the Turing model to domains over $\Re$. Weihrauch, himself a notable contributor to this adapting tradition refutes what I can only call a 'preposterous' claim in [4], p.23 (italics added):

[1] It may appear as if I accept, in this paper, uncritically and unconditionally, the epistemological status of the Church-Turing Thesis as it is currently defined in computability theory. Nothing can be further from the truth, particularly because I prefer to think of the underpinning of the digital computer in terms of constructive mathematics, but increasingly because I am no longer sure that Church ([8]) was faithful to Turing's concept of the subtle differences between machine computablility and human calculability (see the excellent discussion in Hodges, [21]).

> "A major obstacle to reconciling scientific computation and computer science is the present view of the machine, that is *the digital computer*. As long as the computer is seen simply as a *finite* or *discrete* object, it will be difficult to systematize numerical analysis. We believe that the Turing machine as a foundation for real number algorithms can only obscure concepts."

I can only endorse, wholeheartedly, Weihrauch's entirely justifiable claim that the theory presented in his book, [49], p.268, 'refutes [the above] statement.'

But even the gods 'nod', sometimes! Systematizing numerical analysis is one thing; computing over the reals *may well be* quite another thing – especially since the definition of real numbers can be approached from a variety of mathematical and logical points of view. If we concentrate on computing over the reals, then the long and noble analog computing tradition, using only a finite number of discrete objects, can accomplish – at least theoretically – anything that can be achieved by taking the numerical analysis route.

One of the recurrent themes in the work by Smale and his co-workers on *Real Number Computation* ([4]) is the need for a model of computation over the reals so that the classic problems of mathematical physics, or applied mathematics, like those posed by the need to solve, numerically, ordinary differential equations, defined over $\mathbb{R}$, can be achieved in a theoretically satisfactory way. Consider the following system of nonlinear ordinary differential equations, the so-called Rössler System:

$$\frac{dx}{dt} = -\left(y+z\right)$$

$$\frac{dy}{dt} = x + 0.2y$$

$$\frac{dz}{dt} = 0.2 + z\left(x - 5.7\right)$$

Suppose a *General Purpose Analogue Computer* (*GPAS*) is defined in terms of the usual *adders*, *multipliers* and *integrators* as the elementary units, analogous to the elementary operations defining a Turing Machine or a partial recursive function ($\mu-$*recursion*, *minimalization*, etc.,). Then it can be shown that a *GPAS* consisting of 3 adders, 8 multipliers and 3 integrators (the symbolic definitions are given in § 4, below) can simulate the above Rössler System. *The intermediate step of having to use a numerical algorithm to implement a computation, as in a Turing Machine model, is circumvented in the analog computation model.*

Thus, in implementing a particular computation in the Phillips Machine, there is no need for the intermediate step of writing an algorithm for the numerical computation of the dynamics of the differential equation. The differential equation, in turn has the dual purpose of being a model of the economy and that of the dynamics of the machine. In the case of the Phillips Machine a linear differential equation model plays this dual role. However, despite the ingenuity with which Phillips constructed his machine, so that the dynamics of the flows and the stocks can be approximated by linear differential equations, *the actual functioning of the Phillips Machine remains stubbornly nonlinear*.

Is it not better, then, to avoid the linearizations and try to model, directly and faithfully, the nonlinear dynamics of the machine? If this is done, and with the added advantage of circumventing the intermediate step of implementing a numerical algorithm, would not the model dynamics, as represented, in real time, by the analog machine – in our case, the Phillips Machine – depict the actual dynamics of the modelled economy? If this is so, the only problems an analog computing adherent needs to face are those that have been faced, and largely resolved, by the recursion theorist.

This is the reason for seeking the mathematical foundations of analog computation from the point of view of classical computability theory.

The next section is devoted to an introduction to the Phillips Machine with notes on its origins and purpose, albeit in a very general way. In section 3 this is placed in the context of the broader, longer, richer, analog computing tradition in economics – a tradition that stretches at least as far back as Babbage. Section 4 is devoted to issues of a strictly theoretical nature, viewed from the point of view of computability theory. Finally, the concluding section contains some general philosophical, epistemological and methodological speculations on the place of analogue computation within a *Zeitgeist* that seems to be increasingly digital.

2 Notes on the Phillips Machine as an Analog Computer

> "If a single group of equations can be written which defines the *assumed performance* for two separate systems (each of which within itself represents an orderly or definable behavior), one system may be called the complete *analogue* of the other."
>
> [40], p. 557; italics added

In the case of the Phillips Machine, a paradigmatic analog computing device, this reasonable definition leads to a peculiar dissonance. Phillips observes, quite correctly and perceptively, that:

> "The hydraulic model [the Phillips Machine] will give solutions for non-linear systems as easily as for linear ones. It is not even necessary for the relationships to be in analytic form; *so long as the curves can be drawn the machine will record the correct solutions*, within the limits of its accuracy[2]. In giving the equivalent mathematical model, however, the usual linearity assumption will be made,

[2]By this Phillips must mean engineering precision limits to accuracy, as well as, the usual constraints due to the need to respect the laws of the natural sciences in the manufacturing of machine components. 'Limits of its accuracy' has nothing to do with usual finite precision constraints of a digital computer, nor with the problems of numerical analysis (discretizations, etc.).

in view of the difficulty of working with non-linear differential or difference equations."

[29], pp. 287-8; italics added

Why is it necessary to give 'the equivalent mathematical model'? Obviously the definition of analogue, given in [40], above, is predicated upon the construction (or, is it the existence?) of 'a group of equations' defining 'the assumed performance for two separate system'. But is it not possible to define 'analogue computation'[3] in the same sense in which the intuitive notion of *effective calculability* was, eventually, encapsulated in the definition of a Turing Machine? If the analogue computing machine, such as the Phillips Machine, was constructed – from an engineering point of view – on the basis of economic principles, whether in mathematical form or not, then its functioning should mirror the performance of the economy of which it is a 'model'. The intermediary stage of constructing 'a group of equations' raises a whole host of foundational questions – in addition to the serious problem of the meaning of 'model' to be used here – that have not been faced, squarely or otherwise, in the analogue tradition of computation, except in relation to recursion theory.

The 'foundational questions' I am referring to are mostly mathematical and metamathematical in nature. What kind of mathematics should be used in the setting up of the 'group of equations'? In the case of the digital computer there is, by now, a clear, rigorous, answer to a similar question: recursion theory or constructive mathematics. In the case of analogue computation it may appear as if one can rely on standard mathematics for setting up the relevant 'group of equations'. This, however, is an untenable answer, but I will not go into the philosophical or epistemological foundations of the mathematical foundations of analogue computation in this paper.

More importantly, a characteristically perceptive observation made by Richard Goodwin, in his review of Alan Tustin's pioneering servomechanism-based approach to modelling analogue computation in economics, is directly relevant for the issues I am trying to raise:

> "After an introductory defense of using [the very powerful analytical tools of servomechanism engineering] .. in economics, Professor Tustin launches into a lucid exposition of the basic concepts and methods of what has come to be known as the operational calculus. ... It is here that he employs *graphical analysis* with outstanding success. This emphasis on graphics has a special justification in that *economics*, like engineering, *must work with raw material given in the form of empirical curves.*"
>
> [16], p.209; italics added.

[3] I think the inadequacy of the definition in [40] is primarily due to the absence of the qualifying notion of 'computation'. We are, at least in the present context, concerned essentially with computation by analogue machines – just as we would be concerned with computation by digital machines, if we were using Turing Machines, or its approximations.

This is a refreshingly original view and in stark contrast to the indiscriminate assumptions of economic data being defined on, and generated from, uncountable, uncomputable, undecidable domains. The versatility and power of the Phillips Machine, as an analog computer, is precisely the feature that is emphasised here by Goodwin, reflecting the method's importance in Tustin's parallel – servomechanism-based – approach to analogue modelling and analog computation in economics.

The epistemology and philosophy of analogue modelling, in its almost pure mathematical senses, is a question that has to be addressed separately. As far as I am concerned, the Phillips Machine is an analogue computing device – although I am well aware that there was more to its construction than as a repository of mechanisms to facilitate computation.

But why did Phillips decide to build his machine with – mainly – hydraulic components and on hydraulic principles?[4] After all, Phillips himself was a trained, experienced, electrical engineer and he could have built a servomechanism-type of analog computer, if the sole purpose was pure computation. To place the above notes, and this question, in the context of the way Phillips himself admitted were the inspirations for the construction of a *hydraulic* analogue computing device, it may be useful to recall what he wrote on this point:

> "Fundamentally, the problem is to design and build a machine the operations of which can be described by a particular system of equations which it may be found useful to set up as the hypotheses of a mathematical model, in other words, a calculating machine for solving differential equations. Since, however, the machines are intended for exposition rather than accurate calculation, a second requirement is that the whole of the operations should be clearly visible and comprehensible to the onlooker. For this reason hydraulic methods have been used in preference to electronic ones which might have given greater accuracy and flexibility, the machines being made of transparent plastic ('Perspex') tanks and tubes, through which is pumped coloured water.
>
> Both of the models mentioned above deal with macroeconomic theory in terms of money flows; but they are based on an analogy given by Professor Boulding to show how the production flow, stocks and price of a commodity may react on one another."
>
> [29], pp. 283-4; italics added.

Just for the record, I copy here, as Figure 1, the relevant figure and section in Boulding's vintage text on *Economic Analysis* (third edition, [5], p. 107[5]).

[4] In a typically hair-splitting, pointless,discussion about Coddington's use of the phrase 'hydraulic Keynesianism', Patinkin ([28], footnote 8) makes all sorts of allusions and invokes utterly senseless references and correspondence with all and sundry - without taking the trouble, so far as I can infer, to actually read the Phillips paper of 1950. The most astonishing absurdity is a reference to Shackle's phrase '*reduction of economics to hydraulics*' ([36], p.189), and the quite unbelievable conjectures about how he (Shackle) came to construct it.

[5] Phillips refers to p. 117 in an earlier edition of the Boulding book (p. 284).

His co-constructor, Walter Newlyn, in his interesting chapter on *The Origin of the Machine* ([27], pp. 31-2) is more specific:

> "[Phillips'] Figure 3 [in '*Saving and Investment : Rate of Interest and Level of Income'*] shows Boulding's supply and demand hydraulic analogy (1948, p.117) modified for the inclusion of stocks and the interconnections between stocks and flows linked mechanically"[6]

Even a cursory comparison of FIG 2 in [29], p.285 with the figure in Boulding would show that the former is inspired by the latter, in the precise senses in which Phillips and Newlyn suggest. Any perceptive reader of Boulding's suggestive analogy will realize that he is describing *non-steady dynamics*, an euphemism, surely, for *nonlinear dynamics*. It is strange, then, given the aims with which Phillips constructed an analog *hydraulic* computing machine, that he chose to build a model represented by 'a particular system of equations' that were linear.

[6] Incidentally, there is a curious unscholarly 'aside' in this otherwise interesting narrative of 'The Origin of the Machine in a Personal Context', when Newlyn suggests that the 'inclusion' of the accelerator 'in the Mark II machines is the element which gave it the dynamic feature which stimulated the work of Richard Goodwin – a far cry from being a teaching aid.' This unfortunate inaccuracy is triply wrong, First of all, Phillips was inspired by the article on the 'accelerator', by Goodwin, in the Alvin Hansen *Festschrift* ([12]; [29], footnote 1, p.298). Secondly, Goodwin had developed the 'flexible accelerator' model, that which was published as the lead article in the first number of the first issue of the 1951 volume of the **Econometrica**, much earlier than the construction of even the prototype Phillips Machine, in 1949 (cf. [13]). Thirdly, Goodwin had gone much beyond the classic 'flexible accelerator' in his remarkable review of the book by Hicks on the Trade Cycle ([14]). No models by Phillips – including those in his celebrated EJ articles of the 1950s ([30], [31]) – ever managed to encapsulate the richness of Goodwin's nonlinear elements. On the other hand, the Phillips Machine could easily implement the full nonlinear Goodwin model quite easily!

Production, Consumption, and Stocks

We must get a picture of the production and consumption of any given commodity—let us say, wheat—as a process something like the flow of water into and out of a tank. The quantity of water in the tank corresponds to the quantity of the commodity in existence at a moment of time —1,000,000,000 bushels of wheat, say, in the various storehouses of the world. From the production of farms wheat is constantly being added to the total stock; this corresponds to the flow of water into the tank. The flow is not of course regular, for at harvest time it is very large and at other times very small. However, we may call the rate of flow, averaged over a number of years, the *rate of production,* measured not in bushels but in *bushels per year.*

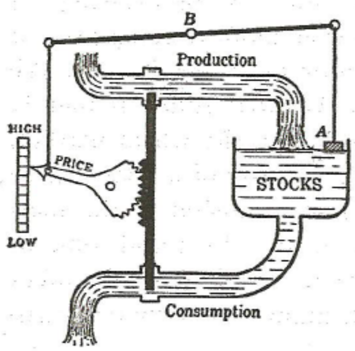

FIG. 9. How Price Regulates Production and Consumption.

Wheat is also continually flowing out into consumption, corresponding to the flow of water out of the tank. This flow may also not be quite regular, though it will be more regular than the flow of production. The average rate of flow may be called the *rate of consumption,* measured also in bushels per year. Fig. 9 may help to visualize this process.

Figure 1: Boulding's Hydraulic Analogy

I should like to end this Introductory section with a few personal remarks on the role of analogue computation and the Phillips Machine in my own intellectual history. I was privileged to have been an engineering student, in the department of mechanical engineering, at Kyoto University, in the transition period when massive digital computing facilities were becoming available and old-fashioned wind tunnels were being replaced by simulations of theoretically derived differential equations on these modern computers. But we were fortunate to have the benefit of the analogue computing tradition being practised and demonstrated in the analysis and studies of nonlinear dynamics by one of the great masters of this method and tool: Chihiro Hayashi ([18]). His lectures on electric circuit theory, with especial emphasis on studying, via analogue computation, the dynamics of the van der Pol equation – in both forced and unforced forms – was my introduction to this method and its versatility and to the famous equation itself. Graphical techniques dominated the way Hayashi approached the study of the intractable forced van der Pol equation – presaging and preparing me for instruction by Richard Goodwin, a few years later, at Cambridge, when this very equation was now given an extraordinary economic interpretation in terms of a Keynesian trade cycle model. Goodwin continued the graphical, geometric, tradition of Hayashi, but also gradually began to move towards representations of the van der Pol equations in digital computing machines.

Remarkably, when I switched to economics, first in the department of economics at the University of Lund, I came under the wise influence of Professor Björn Thalberg, whose work at that time – and for many years earlier and later – was based on the contributions by Goodwin and Phillips to the modelling of a Keynesian vision of the macroeconomy ([43], [44] and [45]. Thalberg had himself attended the very first series of lectures given by Goodwin, at Cambridge, utilising the Cambridge Phillips Machine, in 1952.

Many years later, my external examiner, for the Cambridge PhD thesis that I worked on under Goodwin's enlightened guidance, was Sir Roy Allen, whose successful textbooks of the 1950s, 1960s and 1970s were among the core 'modern' texts, in addition to Samuelson's *Foundations* and Patinkin's *Money, Interest and Prices*, through which I learned macroeconomics in Thalberg's graduate courses, in the early 1970s. No macroeconomic textbook emphasised a servomechanism-based modelling approach to macroeconomics, using, in particular, the Phillips and Goodwin dynamic models, more than Roy Allen's lucid textbooks of that vintage period (see, for example, [2] and [3]).

That my formative years as a macroeconomist were at the feet of Thalberg and Goodwin gave me, what I consider to be an immense advantage over many of the 'modernists': to learn and understand a vision of the macroeconomy in which computation was given crucial interpretive, investigative and theoretical roles. These were the roles that Phillips emphasised, at least in those classic early works and constructions, and Goodwin grappled with all his intellectual life.

3 A Perspective on the Analogue Computing Tradition in Economics

> "The [hydraulic] mechanism just described is the physical analogue of the ideal economic market. The elements which contribute to the determination of prices are represented each with its appropriate rôle and open to the scrutiny of the eye. We are thus enabled not only to obtain a clear and analytical *picture* of the interdependence of the many elements in the causation of prices, but also to employ the mechanism as an instrument of investigation and by it, study some complicated variations which could scarcely be successfully followed without its aid."
>
> ([10], p.44, italics in original)

Calculating, estimating, comparing, constructing and reasoning with numerical ratios, averages, series, tables areas, volumes and so on – in short, 'analyzing data', whether natural or artificial – underpinned much inference, and some deduction – is the way our classical and physiocratic predecessors came to policy precepts. However, with the exception of Charles Babbage and, possibly, Jevons, till Irving Fisher ([10]), in 1891, constructed his 'remarkable hydraulic [analogue computing] apparatus for calculating equilibrium prices' ([6], p.57,) resorting to *actually constructed machine models of computing* in economics seems to have remained an isolated example. Fisher's own description – quoted above – of the functioning of his hydraulic analogue computing machine clarifies an important feature of such computations: *their independence from any intermediation via numerical analysis.*

There were, of course, the famous computing machine *metaphors* used by Walras, Pareto – and, then, inspired by Barone, in the important 'Socialist Calculation Debate', most comprehensively summarised, both critically and constructively[7] by Hayek ([19] & [20]). Lange, returning to the theme over thirty years later, in his *Dobb Festschrift* article on *The Computer and the Market* ([24]), muddied the issue by unscholarly and unsubstantiable claims for the possibilities of a digital computer (having, in the meanwhile, also forgotten that the initial discussions were with reference to *analog computing machines* and, in particular, the metaphor of *the market as an analogue computer*). None of the participants had any technical knowledge of the mathematical underpinnings of computing, in a sense understandably so, since the mathematical foundations of computing were being placed on a rigorous basis just during those very years that the debate was at its height.

Analogue computing techniques in economics had the proverbial still birth. There was a flurry of activities in the late 1940s and early 1950s, quite apart from Phillips, at the hands of Richard Goodwin, Herbert.A. Simon, Robert H.Strotz, Otto Smith, Arnold Tustin, Roy Allen, Oscar Lange and a few others. As we know, now, Phillips built his famous *MONIAC* hydraulic national

[7] Here, the word is not to be interpreted in its mathematical sense!

income machine at the end of the 40s and it was used at many Universities - and even at the Central Bank of Guatemala - for teaching purposes and even as late as the early 70s Richard Goodwin, at Cambridge University, taught me elementary principles of coupled market dynamics using such a machine. Strotz and his associates, at Northwestern University, built electro-analogue machines to study inventory dynamics and nonlinear business cycle theories of the Hicks-Goodwin varieties. Otto Smith and R.M. Saunders, at the University of California at Berkeley, built an electro-analogue machine to study and simulate a Kalecki-type business cycle model. Roy Allen's successful textbooks on Macroeconomics and Mathematical Economics of the 50s - extending into the late 60s - contained pedagogical circuit devices modelling business cycle theories (cf:[2] especially chapter 9; and [3], especially chapter 18). Arnold Tustin's highly imaginative, but failed textbook attempt to familiarise economists with the use of servomechanism theory to build analogue machines as models of economic dynamics ([46]) and Oscar Lange's attractive,elementary, expository book with a similar purpose ([25]) also suffered the fate of 'stillbirth', at the dawn of the digital computing age.

Humphreys ([22]) refers to nonlinear business cycle theories as examples of computational 'studies' that straddle 'the pre-computational era and the era of computational economics', claiming that 'there is no sharp divide between 'the two eras'. This claim can be substantiated by a more finessed study of the particular example of a canonical nonlinear business cycle equation, using – as was, indeed, actually done – analogue computing machines as in the 'pre-computational era' and comparing it with its study using a digital computing machine of the 'era of computational economics'.

The example I have chosen here encapsulates a noble tradition of *computation in economics* in every sense of this concept, to study a precisely specified mathematical system on both analogue and digital computers. It is, in a precise sense, also a substitute for an analytical study (because such a study is provably 'unlikely' to succeed in any meaningful way). Moreover, it can be viewed as an explicit example of an epistemological tool to interpret the results (most of which were unexpected). Finally, to gain insight into the link between a computing machine and *its* theory and the theory of nonlinear dynamical systems. The latter point is turning out to be the most significant from the point of view of the epistemology of computation, since the interaction can only be explored by representing the one system by the other – and, therefore, even an exploration into a new domain: studying the repertoire of digital machine behaviour with analogue computing machines, and *vice versa*.

Consider, therefore, the following equation, representing a classical Keynesian nonlinear multiplier-accelerator model of the dynamics of national income, y:

$$\epsilon \dot{y}(t) + (1-\alpha)\, y(t) = \phi\left[\dot{y}(t-\theta)\right] + \beta(t) + l(t) \tag{1}$$

Now, there are *at least* six different ways to investigate solutions to this nonlinear difference-differential equation:

- In old fashioned analytical modes;
- Using Non-standard analysis;
- Graphically, i.e., in terms of the geometry of dynamic behaviour, as usually done in the qualitative theory of differential equations;
- By the method of equivalent linearization;
- Using an electro-analogue computer;
- Using digital computers;

It is, of course, only the last two alternatives that are of relevance in this discussion. Assuming, for example, $\beta(t) + l(t)$ a constant[8] and reinterpreting $y(t)$ as a deviation from the unstable equilibrium of (1) ($\frac{\beta(t)+l(t)}{(1-\alpha)}$), one obtains a mixed nonlinear difference-differential equation:

$$\epsilon \dot{y}(t+\theta) + (1-\alpha) y(t+\theta) = \phi[\dot{y}(t)] \tag{2}$$

In the first case, expanding (2) by a Taylor series approximation and *retaining only the first two terms*, one obtained the famous (unforced) *Rayleigh* (- *van der Pol*) – *type* equation:

$$\ddot{y} + \left[\chi\left(\frac{\dot{x}}{\dot{x}}\right)\right]\dot{x} + x = 0 \tag{3}$$

With this approximated reformulation began an 'industry' in the endogenous theory of the business cycle, where the cardinal desideratum was the existence of a unique, stable, limit cycle, independent of initial conditions. All four desiderata were violated when the approximations were more precise – in a purely technical sense – and the analysis proceeded via studies by means of analogue and digital computing machines. Even more interestingly, the insights obtained from an analogue computing machine study provided hints in setting up a computing study of (1) by means of digital computing machines.

Now, using an *electro-analog computer*, it was found, in [41], that the approximation of (1) retaining the first four terms of a Taylor series expansion, generated twenty-five limit cycles, and *a potential for a countable infinity of limit cycles* with further higher order terms included in the approximations. Moreover, in its original formulation, one of the desired criteria for the nonlinear formulation of the endogenous model of the business cycle, was to generate self-sustaining fluctuations, *independent of initial conditions*. This latter property was lost when the approximation was made more precise. These aspects are discussed fairly exhaustively in [48].

Next, coupling two equations of type (3), via the *Phillips Electro-Mechanical-Hydraulic Analogue Computing Machine* ([17]), Goodwin and Phillips were able

[8] If $\beta(t) + l(t)$ was not assumed a constant, the obdurate *forced* version of (3) would have to be confronted, wihtout any hope of a disciplined solution even with the help of computing machines, whether analog or digital.

to generate – *unexpectedly* – the *quasi-periodic paradox* (cf., [1]). Neither Goodwin, nor Phillips, who did the coupled-dynamics computation on the *Phillips Machine*, had any clue – theoretical or otherwise – about interpreting and encapsulating this outcome in any economic theoretical formalization. The key point is that they were *surprised* by the outcome and did not know how to interpret it when it *emerged*. This is where the richness of the *epistemology of computation* manifests itself most dramatically. There was no macrodynamic theory to which they could relate the observed behaviour, which was contrary to expected behaviour.

Finally, Zambelli ([50] repeated the exercise in [41] (as did we, in [48]), but this time on a *digital* computer. His results came as a surprise to him: although we can confirm the results in [41], the outcomes are richer and more varied and one would have no idea which way to proceed, if one was wedded to an equilibrium norm to which the results have to conform.

It goes without saying that one of the key differences between analogue and digital computing is that in the latter the intermediation between the continuous and the discrete is achieved by means of numerical procedures; this intermediation is circumvented in the analogue tradition, as pointed out above. In this sense, there *is* a sharp difference between 'the pre-computational era and the era of computational economics'. Much of what is routinely referred to as computational economics in the modern era is simply variations on the theme of numerical analysis, without any anchoring in the mathematical theory of the computer, whether digital or analogue.

There is no better way, at least in my opinion, to end this section than to recall Richard Stone's extraordinarily perceptive reflections on the power and possibilities of analog computing in the kind of multisectoral macrodynamics he was then embarking on, summarised in his talk at a Conference on Automatic Control, as far back as 1951 ([39], pp. 82-3)[9]:

> "Analogue machines are not unknown to economists, but in the past they have been used for demonstration purposes but not for computing. Examples of such machines are the hydrostatic model of general economic exchange described by Irving Fisher in 1892 and subsequently built, and the Phillips-Newlyn Hydraulic model for demonstrating the interdependence of the main variables in general aggregative analysis which was designed a few years ago and which has now been adopted for expository purposes in a number of universities in this country. The use of electric analogues has been suggested recently by various authors, one of these instruments being concerned with the investigation of inventory oscillations and another with the investigation of equilibrium among spatially separated markets. ....

[9] I have suppressed the many references that were in the full quote. Alan Tustin's talk, immediately after Richard Stone, and published back-to-back with Stone's paper ([47]), is also worth a perusal and some reflection, since, by that time, his own important book ([46]) on a servomechanism-based approach to analog computation was close to completion.

> It is impossible to say without detailed investigation whether such an approach to economic computing problems have any advantage over the usual digital methods. Its utility would probably depend on whether a more detailed investigation of economic responses revealed patterns for which a good electric analogue could be devised. Since the machine would be designed to perform calculations arising from a specified system it should be easily modifiable, since otherwise it would be too rigid for the changing circumstances of economic life. It goes without saying that should the design of such a machine be contemplated it would be necessary to introduce many complications which have not been elaborated here. In particular there would be a need to consider (i) stocks as well as flows, (ii) prices and quantities separately instead of simply their product ...., (iii) exogenous factors .... (iv) expectations .... and finally (v) the specification of ..errors. "

Those who are familiar with the Phillips Machine, and its architecture, will know that from the outset Phillips was concerned with issues (i), (ii) and (iv). Knowing his general intellectual interests and subsequent work, there is no doubt at all in my mind that (iii) and (v), too, would have been incorporated in any modern reconstruction of the Phillips Machine.

4 Theoretical Notes on Computability

> "Church's thesis, that all reasonable definitions of 'computability' are equivalent, is not usually thought of in terms of computability by a continuous computer, of which the general-purpose analog computer (GPAC) is a prototype."
>
> [34], p. 1011

Consider the linear, second order, differential equation that once formed the fountainhead of Keynesian, endogenous, macroeconomic theories of the cycle (indeed the kind of equation used in the Phillips differential equation model of the macroeconomy):

$$a\ddot{x} + b\dot{x} + kx = F \tag{4}$$

Solving, as in elementary textbook practice, for the second order term, $\ddot{x}$:

$$\ddot{x} = \frac{1}{a}F - \frac{k}{a}x - \frac{b}{a}\dot{x} \tag{5}$$

Integrating (2) gives the value for $\dot{x}$ to be replaced in the third term in the above equation.

Integrating $\dot{x}$, gives the value for x, and the system is 'solved'[10]. Thus,

three mechanical elements have to be put together in the form of a machine to implement a solution for (2):

- A machine element that would *add* terms, denoted by a circle;
- An element that could *multiply* constants or variables by constants, denoted by an equilateral triangle;
- An element that could '*integrate*', in the formal mathematical sense, without resorting to sums and limiting processes, denoted by a 'funnel-like' symbol;

One adder, three multipliers and two integrators, connected as in Figure 2, can solve the above equation[11]:

Note several distinguishing features of this analogue computing circuit diagram. First of all, there are no time-sequencing arrows, except as an indicator of the final output, the solution, because all the activity, the summing, multiplication and integration, goes on *simultaneously*. Secondly, no approximations, limit processes of summation, etc., are involved in the integrator; it is a natural physical operation, just like the operations and displays on the odometer in a motor car or the voltage meter reading in your home electricity supplier's measuring unit. Of course, there are the natural physical constraints imposed by the laws of physics and the limits of precision mechanics and engineering, something that is common to both digital and analogue computing devices, so long as *physical* realizations of mathematical formalisms are required.

In principle, any ODE can be solved using just these three kinds of machine elements linked appropriately because, using the formula for integrating by parts, a need for an element for differentiating products can be dispensed with. However, these machine elements must be supplemented by two other kinds of units to take into account the usual independent variable, time in most

[10] This is the reason why analogue computing is sometimes referred to as resorting to 'bootstrap' methods. But it is more relevant, especially in computability contexts, to refer to this aspect as 'self-reference'. Recall Goodwin's perceptive observation, more than half a century ago:

> "A servomechanism *regulates its behaviour by its own behaviour* in the light of its stated object and therein lies the secret of its extraordinary finesse in performance. .... It is a matter of considerable interest that Walras' conception of and term for dynamical adjustment - tâtonner, to grope, to feel one's way - is literally the same as that of modern servo theory." (cf.[15] ; italics added)

[11] One must add rules of interconnection such as each input is connected to at most one output, feasibility of feedback connections, and so on. But I shall leave this part to be understood intuitively and refer to some of the discussion in [32], pp. 9-11; observe, in particular, the important remark that (ibid, p.10, italics in the original):

> "*[F]eedback, which may be conceived of as a form of continuous recursion, is permitted.*"

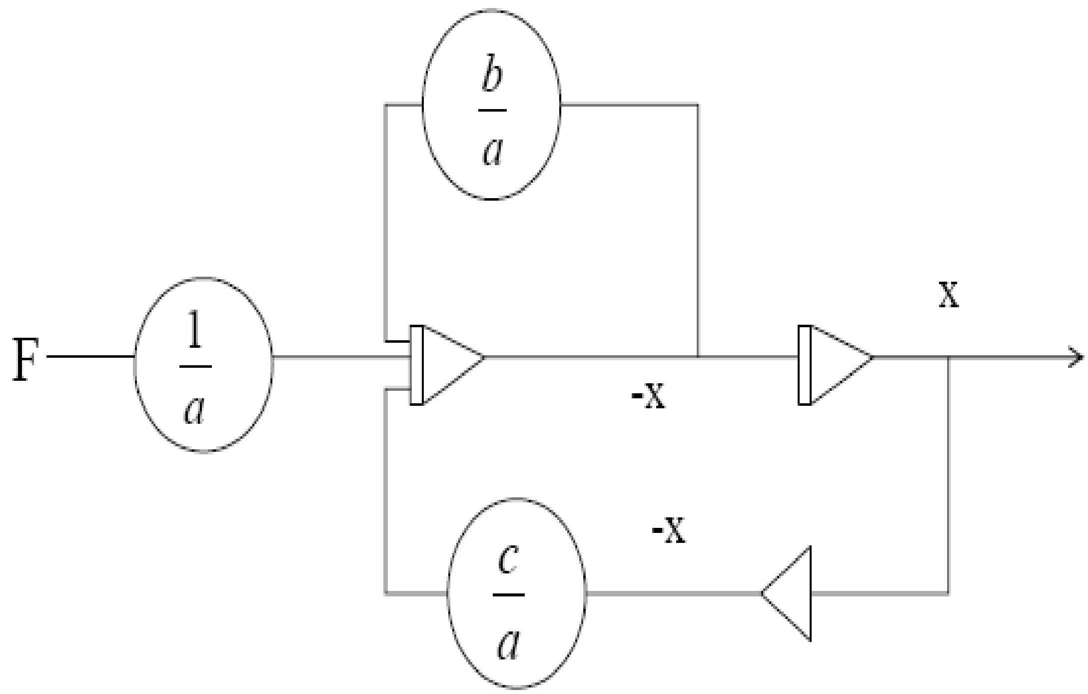

Figure 2: Linear Second Order Differential Equation

cases, and one more to keep track of the reals that are used in the adder and the multiplier. This is analogous to Turing's 'notes to assist the memory', but play a more indispensable role. Just as in Turing's case, one can, almost safely, conclude that 'these elements, appropriately connected, including "bootstrapping" - i.e., with feedbacks - exhaust the necessary units for the solving of an ODE'. Accepting this conjecture *pro tempore*, in the same spirit in which one works within the *Church-Turing Thesis* in *Classical Recursion Theory*, a first definition of an analogue computer could go as follows:

Definition 1 *A General Purpose Analogue Computer (GPAC) is machine made up of the elemental units: adders, multipliers and integrators, supplemented by auxiliary units to keep track of the independent variable and real numbers that are inputs to the machine process, and are interconnected, with necessary feedbacks between or within the elemental units to function simultaneously.*

Recalling the fertile and mutual interaction between partial recursive functions and Turing Machines, one would seek a definition, if possible by construction, of the class of functions that are analog computable by GPACs. These are precisely the *algebraic differential equations* ([32], p.7, [33], p.26, [37], pp.340-3).

Definition 2 *An algebraic differential polynomial is an expression of the form :*

$$\sum_{i=1}^{n} a_i x^{r_i} y^{q_{0i}} \left(y'\right)^{q_{1i}} .... \left(y^{(k_i)}\right)^{q_{k_i i}} \tag{6}$$

where a_i *is a* real *number*, $r_i, q_{0i}, ..., q_{k_i i}$ *are non*−negative integer *valued and* y *is a function of* x.

Definition 3 *Algebraic differential equations (ADEs) are ODEs of the form:*

$$P\left(x, y, y', y'', ..., y^{(n)}\right) = 0 \tag{7}$$

where P is an *algebraic differential polynomial* not identically equal to zero.

Definition 4 *Any solution* $y(x)$ *of an ADE is called differentially algebraic (DA); otherwise they are called transcendentally-transcendental ([33]) or hyper-transcendental ([37]).*

Clearly, the definition of ADEs includes all the usual sets of simultaneous systems of linear and nonlinear differential equations that economists routinely - and non-routinely - use. So, we are guaranteed that they are solvable by means of GPACs. Now one can pose some simple questions, partly motivated by the traditions of classical recursion theory:

- Are the solutions to ADEs, generated by GPACs, computable?

- Is there a corresponding concept to *universal computation* or a universal computer in the case of analogue computation by GPACs?
- Is there a *fix point principle* in analogue computing by GPACs that is equivalent or corresponds to the classic recursion theoretic fix point theorem?
- Is there a '*Church-Turing Thesis*' for analogue computing by GPACs?

The reason I ask just these questions is that an economist who indiscriminately and arbitrarily formulates dynamical hypotheses in terms of ODEs and attempts to theorise, simulate and experiment with them must be disciplined in some way - in the same sense in which recursion theory and numerical analysis discipline a theorist with warnings on solvability, uncomputability, approximability, etc. It is all very well that the Bernoulli equation underpins the Solow growth model or the Riccati equation underpins the use of control theory modelling environments or the Rayleigh, van der Pol and Lotka-Volterra systems are widely invoked in endogenous business cycle theories. Their use for simulations calls forth the conundrums mentioned above for digital computers and they may require other kinds of constraints to be respected in the case of simulations by GPACs. There will, of course, be engineering constraints: precision engineering requirements on the constructions of the adders, multipliers and the integrators can only achieve a certain level of precision, exactly as the thermodynamic constraints of heat irreversibilities in the integrated circuits of the digital computer. I do not attempt to deal with these latter issues in this paper.

The answer, broadly speaking, to the first question is in the affirmative ([32], op.cit, §4, pp.23-27 and [34], Theorems 1 and 1', p.1012).

The answer to the second question is easier to attempt if the question is posed in a slightly different way, in terms of the relation between Turing Machines and Diophantine equations (cf. [26]).

Definition 5 *A relation of the form*

$$D\left(a_1, a_2, ..., a_n, x_1, x_2, ..., x_m\right) = 0 \tag{8}$$

where D is a polynomial with integer coefficients with respect to all the variables $a_1, a_2, ..., a_n, x_1, x_2, ..., x_m$ separated into *parameters* $a_1, a_2, ..., a_n$ and *unknowns* $x_1, x_2, ..., x_m$, is called a *parametric diophantine equation.*

A *parametric diophantine equation*, D, defines a set $\mathcal{F}$ of the parameters for which there are values of the unknowns such that:

$$\langle a_1, a_2, ..., a_n\rangle \in \mathcal{F} \iff \exists x_1, x_2, ..., x_m \left[D\left(a_1, a_2, ..., a_n, x_1, x_2, ..., x_m\right) = 0\right] \tag{9}$$

One of the celebrated mathematical results of the 20th century was the (negative) solution to *Hilbert's Tenth Problem [26].* In the eventual solution of that famous problem two crucial issues were: the characterisation of *recursively* *enumerable sets* in terms of parametric diophantine equations and the relation between Turing Machines and *parametric Diophantine equations*. The former is, for example, elegantly exemplified by the following result ([23], Lemma 2, p.407):

Lemma 6 *For every recursively enumerable set W, there is a polynomial with integer coefficients given by $Q(n, x_1, x_2, x_3, x_4)$, i.e., a parametric diophantine equation, such that, $\forall n \in \mathbb{N}$,*

$$n \in W \iff \exists x_1, \forall x_2, \exists x_3, \forall x_4 \left[Q(n, x_1, x_2, x_3, x_4) \neq 0\right] \tag{10}$$

The idea is to relate the determination of membership in a structured set with the (un)solvability of a particular kind of equation. If, next, the (un)solvability of this particular kind of equation can be related to the determined behaviour of a computing machine, then one obtains a connection between some kind of computability, i.e., decidability, and solvability and set membership. This is sealed by the following result:

Proposition 7 *Given any parametric Diophantine equation it is possible to construct a Turing Machine M, such that M will eventually halt, beginning with a representation of the parametric $n-tuple$, $\langle a_1, a_2, ..., a_n\rangle$ iff (16) is solvable for the unknowns $x_1, x_2, ..., x_m$.*

Suppose we think of ODEs as Parametric Diophantine Equations; recursively enumerable sets as the domain for continuous functions and GPACs as Turing Machines. Can we, then, derive a connection between ODEs, continuous functions and GPACs in the same way as above? The affirmative answer is provided by the following proposition, which I shall call Rubel's Theorem:

Theorem 8 *(Rubel's Theorem): There exists a nontrivial fourth-order, universal, algebraic differential equation of the form:*

$$P(y', y'', y''', y'''') = 0 \tag{11}$$

where P is a homogeneous polynomial in four variables with integer coefficients.

The exact meaning of '*universal*' is the following:

Definition 9 *A universal algebraic differential equation P is such that any continuous function $\varphi(x)$ can be approximated to any degree of accuracy by a C^{∞} solution, $y(x)$, of P. In other words:*

$$\forall \text{ positive continuous } \varepsilon(x),\ \exists y(x)\ s.t\ \mid y(x) - \varphi(x) \mid < \epsilon(x),\ \forall x \in (-\infty, \infty) \tag{12}$$

Recent developments (cf. [9],[7]) have led to concrete improvements in that it is now possible to show the existence of C^n, $\forall n, (3 < n < \infty)$; for example, the following is a specific Universal algebraic differential equation:

$$n^2 y'''' y'^2 - 3n^2 y''' y' + 2n\,(n-1)\, y''^2 = 0 \tag{13}$$

In this sense, then, there is a counterpart to the kind of universality propositions in classical recursion theory – *computation universality*, *universal computer*, etc., – also in the emerging theory for analogue computation, particularly, GPACs. Eventually, by directly linking such universal equations to Turing Machines via numerical analysis there may even be scope for a more unified and encompassing theory.

As for the third question, my answer goes as follows. GPACs can also be considered *generalised fix-point machines*! Every solution generated by a GPAC is a fixed-point of an ADE. This is a reflection of the historical fact and practice that the origins of fixed point theory lies in the search for solutions of differential equations, particularly ODEs[12].

Whether there is a Church-Turing Theses for analogue computation is difficult to answer. The reason is as follows. The concept of computability by finite means was made formally concrete after the notions of solvability and unsolvability or, rather, decidability and undecidability, were made precise in terms of recursion theory. These notions were made precise within the context of a particular debate on the foundations of mathematics - on the nature of the logic that underpinned formal reasoning. As Gödel famously observed:

> "It seems to me that [the great importance of general recursiveness (Turing's computability)] is largely due to the fact that with this concept one has for the first time succeeded in giving an absolute definition of an interesting epistemological notion, i.e., one not depending on the formalism chosen. In all other cases treated previously, such as demonstrability or definability, one has been able to define them only relative to a given language, and for each individual language it is clear that the one thus obtained is not the one looked for. For the concept of computability however, although it is merely a special kind of demonstrability or decidability[13] the situation is different. By a kind of miracle it is not necessary to

[12] But also PDEs (partial differential equations), as George Temple pointed out ([42], p.119):

> "One of the most fruitful studies in topology has considered the mapping T of a set of points S into S, and the existence of fixed points x such that
>
> $$Tx = x$$
>
> The importance of these studies is largely due to their application to ordinary and partial differential equations which can often be transformed into a functional equation $Fx = 0$ with $F = T - I$ where $Ix = x$."

[13] I have always wondered whether this is not a misprint and the word that is meant to be here is not 'decidability' but 'definability'!

distinguish orders, and the diagonal procedure does not lead outside the defined notion. This, I think, should encourage one to expect the same thing to be possible also in other cases (such as demonstrability or definability)."

[11], p.84.

So, to ask and answer an epistemological question such as whether there is a correspondence to the 'Church-Turing Thesis' in analogue computing by GPACs must mean that we must, first, characterise the formal structure and mathematical foundations of ODEs in a more precise way. I think this is an interesting methodological task, but cannot even be begun to be discussed within the confines of a simple expository paper such as this. I think, however, there will be an interplay between a logic on which continuous processes can be underpinned, say by Lukasiewicz's *continuous logic*, and the logic of ODEs[14]. My intuition is that there will be some kind of 'Church-Turing Thesis' in the case of analogue computing by GPACs and awareness of it will greatly discipline solution, simulation and experimental exercises by the use of GPACs (see also [34]).

5 Analogue Reflections in a Digital World

"[I]t is, therefore, desirable to form some view of the degree of complexity that may be expected in a 'scheme of dependence' (or a 'model' as the economist calls it) such as would be adequate as a basis for prediction and the analysis of requirements for stabilisation.

...

My own conclusion is that the representation by a reasonably accurate *analogue* of any of the schemes of dependence that have so far been proposed by economists presents no inherently insuperable problems, though it provides a sufficient spice of difficulty to make the problem interesting. I think it is for the economists to say whether such a project, if it were realised, would be of value. If it were I am sure that there are many engineers skilled in the field of *analogue devises* who would be glad to help transform the suggestion into reality."

[47], pp. 85, 89; italics added.

In the intervening six decades, or so, since Tustin's 'conclusion' and 'suggestion', the economist has abandoned the engineer and pawned the subject to the mathematician – and not just any mathematician, but to the kinds whose foundations are most alien to computation. How can the economist reclaim the

[14] I suspect that this will be fruitful link to pursue partly because Lukasiewicz, in the development of his continuous valued logic, abandons both the law of the excluded middle and proof by the method of *reductio ad absurdum* - both contentious issues in the debate between Hilbert and Brouwer that led to the foundational crisis in mathematics from which the work of Gödel and Turing emerged.

analogue vision, to return to thinking in terms of 'schemes of dependence', and to construct machine models that place at the centre of concern 'the analysis of requirements for stabilisation', without ideological anchorings in a political economy of nihilism?

Paradoxically, my own diagnosis of the malaise in current mathematical economics is rooted in a critique of its recent obsession with computation and the halo surrounding numerical analysis. Computable General Equilibrium theory lies at the core of the frontier research in macroeconomics in its guise as Recursive Macroeconomics, whose 'scheme of dependences' are formalised as Stochastic Dynamic General Equilibrium schemes. Another strand of popular research at the frontiers is the so-called agent-based economic 'scheme of dependences', again underpinned by – and in – atheoretical computation schemes, but, in reality, disciplined by (unconsciously) recursion theory. Varieties of experimental economics, algorithmic game theory and a loose amalgam of subjects under the umbrella phrase 'computational economics' are all trying to do the impossible – i.e., to compute the uncomputable, decide the undecidable and complete the incompleatable.

Why?

Only because there is an unbridgeable gap between the mathematics in which the above economic sub-disciplines are theorized and that in which their computational lives are implemented. If you theorise in terms of real analysis, founded on set theory plus the axiom of choice, but compute with the aid of a digital computer, then the mathematical foundations of the latter – recursion theory or constructive mathematics – creates the inevitable dissonance.

Surely there are two obvious ways to overcome this dissonance. One, to return economic theory to its algorithmic roots and theorise in terms of constructive or computable mathematics, *ab initio.* Two, to retain the rich harvest of results that have been obtained in economic theory, in the almost two and a half centuries of sustained effort by a galaxy of economic theorists, by means of returning to the tradition of formulating 'schemes of dependence', free of allegiance to any kind of mathematics, and build analogue devices to represent them in action.

This is the message I infer from the pioneering work, and aims, that went into the construction, and operation, of the **Phillips Machine**.

There are eminent applied mathematicians and computer scientists who have been seeking a model of computation that can resolve a different dissonance: that between computability theory and numerical analysis. Smale and his co-workers are a distinguished example of this group. In their defining work, [4], they emphasise the following distinctions between computability theory and numerical analysis:

> "There is a substantial conflict between theoretical computer science and numerical analysis. These two subjects with common goals have grown apart. For example, computer scientists are uneasy with calculus, whereas numerical analysis thrives on it. On the other hand numerical analysts see *no use for the Turing machine.*

The conflict has at its roots another age-old *conflict*, that *between the continuous and the discrete.* Computer science is oriented by the digital nature of machines and by its discrete foundations given by Turing machines. For numerical analysis, systems of equations and differential equations are central and this discipline depends heavily on the continuous nature of the real numbers. ...

Use of Turing machines yields a unifying concept of the algorithm well formalized. ....

The situation in numerical analysis is quite the opposite. Algorithms are primarily a means to solve practical problems. *There is not even a formal definition of algorithm in the subject.* ....

A major obstacle to reconciling scientific computation and computer science is the present view of the machine, that is, the digital computer. As long as the computer is seen simply as a finite or discrete subject, it will be difficult to systematize numerical analysis. We believe that the Turing machine as a foundation for real number algorithms can only obscure concepts.

Towards resolving the problem we have posed, we are led to expanding the theoretical model of the machine to allow real numbers as inputs."

[4], p.23; italics added.

Unfortunately, they fail to point out that also in constructive mathematics 'there is not even a formal definition of algorithm in the subject'; yet it can act as the mathematical foundation for the digital computer without any of the conundrums that an intermediary role for numerical analysis calls forth. But most paradoxically, constructive mathematics can also act as an adequate mathematical foundation for analog computation!

I think the 'conflict' Smale and others see between computer science and numerical analysis, between the continuous and the discrete, between the finite and the infinite, are, in my opinion artificial phantoms, created by the unholy alliance between the metaphysics of the real number system and the spectre of numerical analysis.

There is no better way to summarise the main theme in this paper than in terms of the hilarious wisdom of Terry Pratchett in **Making Money, p. 63** (italics added)**:**

"The Glooper[15], as it is affectionately known, is what I call a quote 'analogy machine' unquote. *It solves problems not by considering then as numerical exercise but by actually duplicating them in a form we can manipulate*: in this case, the flow of money and its effects within our society becomes water flowing through a glass matrix, the Glooper."

[15] If my reading of **Making Money** is not entirely 'off base', *The Glooper* **is** the *Phillips Machine*, of sorts!

References

[1] Abraham, Ralph (1985), *Is There Chaos Without Noise*, **Chaos, Fractals, and Dynamics** edited by P. Fischer & William R. Smith, chapter 7, pp. 117-121, Marcel Dekker, Inc., New York & Basel.

[2] R.G.D.Allen (1959), **Mathematical Economics**, Second Edition, Macmillan, London.

[3] R..D Allen (1967), **Macro-Economic Theory: A Mathematical Treatment**, Macmillan, London, 1967.

[4] Blum, Lenore, Felipe Cucker, Michael Shub and Steve Smale (1998), **Complexity and Real Computation**, Springer Verlag, New York.

[5] Boulding, Kenneth. E (1955), **Economic Analysis** (Third Edition), Harper and Row, New York.

[6] Brainard, William. C & Herbert. E Scarf (2005), *How to Compute Equilibrium Prices in 1891*, in: **Celebrating Irving Fisher - The Legacy of a Great Economist**, edited by Robert W. Dimand and John Geanakoplos, Blackwell Publishing Ltd., Oxford.

[7] Keith M. Briggs (2000), *Another Universal Differential Equation*, **tantalum.bt-sys.bt.co.uk:tex/universal-ode.tex**, April 13, 2000.

[8] Church, Alonzo, (1937), *Review of Turing 1936*, **Journal of Symbolic Logic**, Vol. 2, No 1, March, pp.42-43.

[9] R.J.Duffin (1981), *Rubel's Universal Differential Equation,* **Proceedings of the National Academy of Sciences USA**, Vol. 78, No.8, August, pp. 4661-2.

[10] Fisher, Irving (1892; 1991), **Mathematical Investigations in the Theory of Value and Prices**, Augustus M. Kelley, Publishers, Fairfield, NJ.

[11] Kurt Gödel [1946, (1965)], *Remarks Before the Princeton Bicentennial Conference on Problems in Mathematics*, in **The Undecidable: Basic Papers on Undecidable propositions, Unsolvable Problems and Computable Functions** edited by Martin Davis, Raven Press, New York, pp. 84-88.

[12] Goodwin, Richard M (1948), *Secular and Cyclical Aspects of the Multiplier and the Accelerator*, in: **Income, Employment and Public Policy: Essays in Honour of Alvin H. Hansen**, pp. 108-132, W.W. Norton & Company, Inc., New York.

[13] Goodwin, Richard M (1949), *The Business Cycle as a Self-Sustaining Oscillator*, **Econometrica**, Vol. XVII, #2, April, pp. 184-5.

[14] Goodwin, Richard M (1950), *A Non-Linear Theory of the Cycle*, **The Review of Economics and Statistics**, Vol. XXXII, #4, Nov., pp. 316-20.

[15] Goodwin, Richard. M (1951), *Iteration, Automatic Computers, and Economic Dynamics*, **Metroeconomica**, Vol. III, Fasc. 1, April, pp. 1-7.

[16] Goodwin, Richard M (1955), *Review* of: The Mechanism of Economic Systems: An Approach to the Problem of Economic Stabilization from the Point of View of Control-System Engineering, **Review of Economics and Statistics**, Vol. XXXVII, #2, May, pp. 209-10.

[17] Goodwin, Richard. M (2000), *A Superb Explanatory Device*, in: **A.W.H. Phillips - Collected Works in Contemporary Perspective**, edited by Robert Leeson, chapter 13, pp. 118-9, Cambridge University Press, Cambridge.

[18] Hayashi, Chihiro (1964), **Nonlinear Oscillations in Physical Systems**, Princeton University Press, Princeton, New Jersey.

[19] Hayek, F. A von (1935), **Collectivist Economic Planning - Critical Studies on the Possibilities of Socialism**, edited, with an Introduction and a Concluding Essay by F. A von Hayek, Routledge & Kegan Paul, Ltd., London.

[20] Hayek, F. A von (1940), *Socialist Calculation: The Competitive 'Solution'*, **Economica**, N.S., Vol. 7, No. 26, May, pp. 125-149.

[21] Hodges, Andrew (2008), *What Did Alan Turing mean by 'Machine'?*, chapter 5, pp. 75-90, in **The Mechanical Mind in History**, edited by Philip Husbands, Owen Holland, and Michael Wheeler, The MIT Press, Cambridge, Massachusetts.

[22] Humphreys, Paul (2009), *Computational Economics*, Chapter 13, in: **The Oxford Handbook of Philosophy of Economics**, edited by Harold Kincaid & Don Ross, Oxford University Press, Oxford.

[23] James Jones (1981), *Classification of Quantifier Prefixes over Diophantine Equations*, **Zeiechrift für Mathematische Logik und Grundlagen der Mathematik**, Vol. 27, pp. 403-10.

[24] Lange, Oskar (1967), *The Computer and the Market*, in: **Socialism, Capitalism & Economic Growth - Essays presented to Maurice Dobb**, edited by C. H. Feinstein, pp. 158-161; Cambridge University Press, Cambridge.

[25] Oskar Lange (1970), **Introduction to Economic Cybernetics**, Pergamon Press, London.

[26] Yuri V. Matiyasevich (1993), **Hilbert's Tenth Problem**, The MIT Press, Cambridge, Massachusetts.

[27] Newlyn, Walter (2000), *The Origin of the Machine in a Personal Context*, in: **A.W.H. Phillips - Collected Works in Contemporary Perspective**, edited by Robert Leeson, Chapter 8, pp. 31-8, Cambridge University Press, Cambridge.

[28] Patinkin, Don (1990), *On Different Interpretations of the General Theory*, **The Journal of Monetary Economics**, Vol. 26, October, pp. 205-43.

[29] Phillips, A.W (1950), *Mechanical Models in Economic Dynamics*, **Economica**, New Series, Vol. 17, # 67, August, 1950; pp. 283-305)

[30] Phillips, A. W (1954), *Stabilisation Policy in a Closed Economy*, **The Economic Journal**, Vol. 64, #254, June, pp.294-323.

[31] Phillips, A. W (1957), *Stabilisation Policy and the Time-Forms of Lagged Responses*, **The Economic Journal**, Vol. 67, #266, June, pp.265-277.

[32] Marian Boykan Pour-El (1974), *Abstract Computability and Its Relation to the General Purpose Analog Computer (Some Connections Between Logic, Differential Equations and Analog Computers)*, **Transactions of the American Mathematical Society**, Volume, 199, November, PP. 1-28.

[33] Lee A. Rubel (1988), *Some Mathematical Limitations of the General-Purpose Analog Computer*, **Advances in Applied Mathematics**, Vol. 9, pp.22-34.

[34] Lee A. Rubel (1989), *Digital Simulation of Analog Computation and Church's Thesis*, **Journal of Symbolic Logic**, Vol. 54, No.3, September, pp. 1011-17.

[35] Tim Sauer and James A. Yorke (1991), *Rigorous Verification of Trajectories for the Computer Simulation of Dynamical Systems*, **Nonlinearity**, Vol. 4, pp. 961-79.

[36] Shackle, G.L.S (1967), **The Years of High Theory: Invention & Tradition in Economic Thought 1926-1939**, Cambridge University Press, Cambridge.

[37] Claude E. Shannon (1941), *Mathematical Theory of the Differential Analyzer,* **Journal of Mathematics and Physics**, Vol. XX, No. 4, December, pp. 337-354.

[38] Smale, Stephen, (1991), *Theory of Computation*, pp. 60-9, in: **Mathematical Research Today and Tomorrow: Viewpoints of Seven Fields Medalists!**, edited by C.Casacuberta and M. Castellet, Springer-Verlag, Berlin and Heidelberg.

[39] Stone, Richard (1951-2), *Transaction Models, Information and Computing*, **The Review of Economic Studies**, Vol. 19, #2, pp.67-84.

[40] Strotz, R. H, J. F. Calvert & N.F. Morehouse (1951), *Analogue Computing Techniques Applied to Economics*, **AIEE Transactions,** Volume 70 (1), pp. 557-563.

[41] Strotz, R.H, J.C. McAnulty & J. B. Naines, Jr., (1953), *Goodwin's Non-linear Theory of the Business Cycle: An Electro-Analog Solution*, **Econometrica**, Vol. 21, No. 3, July, pp. 390-411.

[42] George Temple (1981), **100 Years of Mathematics**, Gerald Duckworth & Co. Ltd., London.

[43] Thalberg, Björn (1966), **A Trade Cycle Analysis: Extension of the Goodwin Model**, Studentlitteratur, Lund.

[44] Thalberg, Björn (1971), *Stabilization Policy and the Non-Linear Theory of the Trade-Cycle,* **The Swedish Journal of Economics**, Vol. 73, No. 3 September, pp. 294-310.

[45] Thalberg, Björn, (1971), *A Note on Phillips' Elementary Conclusions on the Problems of Stabilization Policy*,**The Swedish Journal of Economics**, Vol. 73, No. 4 December, pp. 385-408

[46] Tustin, Arnold (1953), **The Mechanism of Economic Systems: An Approach to the Problem of Economic Stabilisation from the Point of View of Control-System Engineering**, Harvard University Press, Cambridge, Massachusetts.

[47] Tustin, Arnold (1951-2), *An Engineer's View of the Problem of Economic Stability and Economic Regulation*, **The Review of Economic Studies**, Vol. 19, #2, pp. 85-89.

[48] Velupillai, K. Vela & N. Dharmaraj (2011), *The Time-to-Build Tradition in Business Cycle Modelling*, **ASSRU Discussion Paper**, 09-11, March, Trento.

[49] Weihrauch, Klaus (2000), **Computable Analysis: An Introduction**, Springer-Verlag,Berlin & Heidelberg.

[50] Zambelli, Stefano (2010), *Flexible Accelerator Economic Systems as Coupled Oscillators*, Forthcoming in: the **Journal of Economic Surveys,** Vol. XXV, #1, (2011).